REMARKS

I. Introduction

In response to the Office Action dated August 11, 2004, claims 1, 11, and 21 have been amended, and 27-29 have been added. Claims 9, 10, 19, and 20 have been withdrawn from consideration. Claims 1-8, 11-18, and 21-29 remain in the application for consideration. Re-examination and re-consideration of the application, as amended, is requested.

II. Claim Amendments

Applicant's attorney has made amendments to the claims as indicated above. These amendments were made solely for the purpose of clarifying the language of the claims, and were not required for patentability or to distinguish the claims over the prior art.

III. Interview Summary

On August 2, 2004, an interview was conducted by Examiner Singh and Applicant's attorney, Jason S. Feldmar. A restriction requirement to be issued by the Examiner was discussed. Agreement was reached in that claims 1-8, 11-18, and 21-26 were elected with traverse.

IV. Election/Restriction Requirement

At pages 2-3 of the Office Action, the Examiner required restriction of the application to one of 2 allegedly distinct inventions:

Group I: Claims 1-8, 11-18, and 21-26, drawn to rendering the next frame of the current scene; estimating a bandwidth availability for texture transfer; identifying the amount of data in a texture required for rendering a next scene; splitting said required texture into texture portions that satisfy said bandwidth availability; and transferring one of said texture portion from said data storage means to said texture storage means, classified in class 345, subclass 582.

Group II: Claims 9-10, 19 and 20, drawn to rendering the next frame of the current scene; identifying a time before the rendering for the next frame begins; comparing said time identified as above with the time required to delete a texture from said texture storage means; and if it can be completed before the next frame rendering is due to begin, deleting a texture from said texture storage means, classified in class 345, subclass 552.

As discussed in the interview with the Examiner on August 2, 2004, Applicant elects Group I, namely claims 1-8, 11-18, and 21-26, with traverse.

35 U.S.C. §121 provides that "If two or more independent and distinct inventions are claimed in one application, the Commissioner may require the application to be restricted to one of the inventions." M.P.E.P. §802.01 deviates from the plain meaning of "independent and distinct" by interpreting "and" to mean "or". The Patent Office relies on the absence from the legislative history of anything contrary to this interpretation as support for their position that "and" means "or". Applicants respectfully note that this position is contrary to the rules of statutory construction. Restriction between two dependent inventions is not permissible under the plain meaning of 35 U.S.C. §121.

The Examiner does not assert that the inventions of the claim groups listed above are independent. Rather, the Examiner alleges that the inventions of the claim groups listed above are distinct because they are separately usable in that invention I has separate utility such as to keep texture data to move in an efficient manner based on bandwidth availability by splitting texture data into portions to match the bandwidth available without requiring invention II which is about identifying a time before a next frame begins and comparing said time identified as above with the time required to delete a texture accordingly. Applicants assert that restriction is improper because both claim groups address rending animated image data in real time. In this regard, both claims address the issues of storing texture data in texture storage means. While claim group I addresses the transfer of data to the texture storage means, claim group II merely addresses the deletion of data from the texture storage means to make room for the data transferred to the storage means. Accordingly, Applicant urges the Examiner take into consideration that the subject matter of each of the claim groups is linked by this common inventive concept that utilizes bandwidth to transfer material to from texture storage means.

In addition, Applicant submits that dependent claims 5-6, 15-16 and 24 from Group I provide similar limitations to that set forth in Group II. According to M.P.E.P. §803, there are two criteria for a proper restriction requirement. First, the two inventions must be independent and distinct. In addition, there must be a serious burden on the Examiner if restriction is not required. Even if the first criterion has been met in the present case, which it has not, the second criterion has not been met. Applicants assert that a search into prior art with regard to the invention of the

different groups is so related that separate significant search efforts should not be necessary. In this regard, a search effort regarding dependent claims 5-6, 15-16, and 24 would be nearly identical to the search for prior art for the claims in Group II. Accordingly, there is no serious burden on the Examiner to collectively examine the claim groups listed above. Therefore, restriction is not proper under M.P.E.P. §803.

V. Prior Art Rejections

In paragraphs (5)-(6) of the Office Action, claims 1-5, 7, 8, 11-15, 17, 18, 21-23, 25, and 26 were rejected under 35 U.S.C. §103(a) as being unpatentable over Griffin et al., U.S. Patent No. 5,880,737 (Griffin) in view of Deering et al., U.S. Patent No. 6,535,220 (Deering). In paragraph (7) of the Office Action, claims 6, 16, and 24 were rejected under 35 U.S.C. §103(a) as being unpatentable over Griffin in view of Deering as applied to claim 1, and further in view of Morioka, U.S. Patent No. 6,677,955 (Morioka).

Specifically, the independent claims were rejected as follows:

> Regarding claim 1, Griffin et al. teaches central processing means (processor 132, Fig. 2), data storage means (main memory 134, Fig. 2; shared memory 216, Fig. 6), graphical processing means (image processing hardware 144, Fig. 2), and texture storage means (texture cache 402, Fig. 9A; texture memory 664, Fig. 16). Image data is stored in data storage means (main memory 134, Fig. 2; shared memory 216, Fig. 6). Texture data is stored in texture cache 402 and the data is so organized so that 16 texture elements can be accessed every clock cycle. Col. 2, lines 21-67 discloses generation of new display image every fraction of a second similar to rendering the next frame of the current scene. Griffin et al. discloses the memory bandwidth limitation placed on texture mapping (col. 2, lines 46-48) and identifying and splitting of required texture data into a texture block references which are then placed in a texture cache for retrieval as necessary. Griffin et al. is silent about estimating a bandwidth availability for texture transfer that is unlikely to interfere with the real-time rendering of the current scene. Deering et al. discloses a graphics system 100 comprising a rendering engine 102, texture memory 103, sample buffer 104 and filtering engine 104. It further discloses rendering engine being pushed beyond the limit of its processing capacity and collecting statistics on one or more previous frames and using smaller values for pixel array sizes before sending the primitive data for the current frame (...a controlling agent e.g. processor...may be configured to gather performance measurements...and to generate an initial estimate of frame rendering time...col. 2, lines 40-59; col. 5, lines 1-67; col. 6, lines 1-8). Therefore, it would have been obvious to a person of ordinary skill in the art at the time invention was made to modify the device as taught by Griffin et al. with the feature "checking the capacity or bandwidth availability of a rendering engine and using smaller pixel array values based on such determination" as taught by Deering et al. because it results in faster and improved quality of the displayed video output.
>
> Regarding claims 11, it is similar in scope to claim 1 above and is rejected under the same rationale.
>
> Regarding claim 21, it is similar in scope to claim 1 above and is rejected under the same rationale.

Applicant traverses the above rejections for one or more of the following reasons:

(1) Griffin, Deering, and Morioka do not teach, disclose or suggest pre-fetching texture data into texture storage means;

(2) Griffin, Deering, and Morioka do not teach, disclose or suggest estimating a bandwidth availability for texture transfer that is unlikely to interfere with the real-time rendering of the current scene; and

(3) Griffin, Deering, and Morioka do not teach, disclose or suggest transferring a portion of texture data required for rendering a next scene into texture storage means using available bandwidth while rendering a current scene.

Independent claims 1, 11, and 21, and are generally directed to rendering animated image data in real time. The data rendered for a scene contains texture data that must be transferred to texture storage prior to rendering a scene. Various frames of a current scene are rendered. While rendering the current scene, bandwidth may be available for transferring the texture data for a subsequent scene. Accordingly, the claims provide the ability to estimate the bandwidth availability that can be used to transfer texture data that is unlikely to interfere with the real-time rendering of the current scene. The amount of texture data needed for the next scene is identified and split into various texture portions, the size of which are based on the bandwidth availability. Once split, the claims provide for transferring one of the split portions into texture storage using the available bandwidth while rendering, in real-time, the current scene.

The cited references do not teach nor suggest these various elements of Applicant's independent claims.

Griffin merely describes a system for accessing texture data in a graphics rendering system allows texture data to be stored in memories with high latency or in a compressed form. Griffin's system utilizes a texture cache to temporarily store blocks of texture data retrieved from an external memory during rendering operations. In one implementation, geometric primitives are stored in a queue long enough to absorb the latency of fetching and possibly decompressing a texture block. The geometric primitives are converted into texture block references, and these references are used to fetch texture blocks from memory. A rasterizer rasterizes each geometric primitives as the necessary texture data becomes available in the texture cache. In another of Griffin's implementations, geometric primitives are converted into pixels, including a pixel address, color data, and a texture request. These pixels are stored in a queue long enough to absorb the latency of

a texture block fetch. The texture requests are read from the queue and used to fetch the appropriate texture blocks. As texture data becomes available in the texture cache, the texture data is sampled as necessary and combined with the pixel data read from the queue to compute output pixels.

However, Griffin completely lacks any discussion about pre-fetching texture data or transferring texture data into texture storage for an upcoming scene while rendering a current scene in real time. In this regard, instead of prefetching texture data, Griffin describes merely placing texture data in a queue that is used to supplement a texture cache and absorb the latency caused by accessing texture data in a buffer (see col. 17, lines 19-35). Further, Griffin fails to determine any bandwidth availability as set forth in the claims. In addition, Griffin does not teach or suggest, implicitly or explicitly, transferring a portion of texture data into texture storage means using determined available bandwidth while rendering a current scene.

The Office Action agrees that Griffin fails to teach all of the claim elements (see page 4 of the Office Action) and relies on Deering to disclose the estimation of bandwidth availability as claimed. However, Applicant submits that Deering fails to cure the deficiencies of Griffin.

Deering merely describes a graphics system comprising a texture memory, a rendering engine, a sample buffer and a filtering engine (see Abstract). Deering addresses the situation when image quality suffers as the ratio of output pixels to render pixels drops. In this regard, Deering describes the prior art methods that drop frames which is considered a worse outcome than lowering the image quality for a short while (see col. 1, lines 34-38). To solve the prior art problem, Deering merely describes reducing the resolution of several frames for a short time (see col. 2, lines 40-58). As part of this reduction, Deering determines if the number of primitives for a current frame is anticipated to be too large. If so, Deering commands the rendering engine to use smaller values for the render pixels before sending the primitive data for the current frame (i.e., the image quality is reduced (see col. 5, line 28 – col. 6, line 8).

However, Deering does not even remotely describe a determination of bandwidth availability for transferring texture data that is unlikely to interfere with real-time rendering of the current scene. Instead, Deering merely determines if the primitives of the current frame are too large to maintain rendering in real-time. Such a determination completely fails to address or even allude to available bandwidth that can be used for transferring additional data such as texture data. Further, the result

of Deering's evaluation is used solely for the purpose of reducing the image quality of the current scene being rendered. In this regard, Deering does not teach, suggest, contemplate, or imply, the transfer of texture data using bandwidth that is available while rendering the current scene. Any such determination and transfer is completely outside of the scope and teaching of both Griffin and Deering.

In addition to the above, the Office Action submits that Griffin can be combined with Deering because it results in faster and improved quality of the displayed video output. Applicant respectfully disagrees. Firstly, Deering may improve the overall display over time but explicitly reduces the quality of the image rather than dropping frames (see col. 1, lines 34-38 and 49-53 and col. 2, lines 40-58). Such a reduced quality is completely contrary to improving the quality of the frames in a scene. Further, the combination of Deering with Griffin would still fail to teach the invention as claimed. For example, Deering examines the texture primitives and determines if they are too large for the rendering engines processing capability (see col. 5, lines 5-27). In other words, it may be suggested that Deering determines if a latency (caused by slow processing by the rendering engine) in using the texture data exists. Such a latency is also recognized in Griffin. However, to solve the problem, Deering reduces the quality of the rendered data while Griffin merely places the texture data in a queue. Accordingly, the combination would merely provide for both using smaller values for rendering pixel values (in accordance with Deering) and placing the texture data in a queue (in accordance with Griffin). Such a combination fails to teach estimating the bandwidth availability and transferring data for a <u>next scene</u> using the available bandwidth while rendering the <u>current scene</u> as claimed.

In addition to the above, the remaining cited references also fail to cure the deficiencies of Griffin and Deering.

Moreover, the various elements of Applicant's claimed invention together provide operational advantages over Griffin, Deering, and Morioka. In addition, Applicant's invention solves problems not recognized by Griffin, Deering, and Morioka.

Further, Applicant has added new claims 27-29 that further elaborate on the available bandwidth. Namely, the available bandwidth that is used to transfer the texture data is the data between the rendering of frame in the current scene and a buffer swap used to render a subsequent

frame in the current scene. Such a use of available bandwidth is not described, anticipated, or rendered obvious by the cited references.

Thus, Applicant submits that independent claims 1, 11, and 21 are allowable over Griffin, Deering, and Morioka. Further, dependent claims 2-8, 12-18, and 22-29 are submitted to be allowable over Griffin, Deering, and Morioka in the same manner, because they are dependent on independent claims 1, 11, and 21, respectively, and thus contain all the limitations of the independent claims. In addition, dependent claims 2-8, 12-18, and 22-29 recite additional novel elements not shown by Griffin, Deering, and Morioka.

VI. Conclusion

In view of the above, it is submitted that this application is now in good order for allowance and such allowance is respectfully solicited. Should the Examiner believe minor matters still remain that can be resolved in a telephone interview, the Examiner is urged to call Applicant's undersigned attorney.

Respectfully submitted,

GATES & COOPER LLP
Attorneys for Applicant(s)

Howard Hughes Center
6701 Center Drive West, Suite 1050
Los Angeles, California 90045
(310) 641-8797

Date: November 12, 2004

By:
Name: Jason S. Feldmar
Reg. No.: 39,187

JSF/